Assignment #6 - Simple Web Chat Service

Introduction

The assignment consisted of changing the previous Chat App so that the app could send messages to a web server through http requests instead of using sockets to talk to another app.

Development

All the steps given in the assignment were followed and the class slides were used as guidelines for project organization (packaged and classes' names conventions). The rubric was fulfilled and also used as a guideline of the most important requirements for the system.

The classes RequestService, ServiceHelper and RequestProcessor were created and stored inside the package **requests**. All the other auxiliary classes were also stored inside the same package, for maintainability purposes.

An error was found in the class slides regarding the name of the HTTP Request Header "Content-Type". In the slides, the mentioned header was written as "CONTENT_TYPE", with an underscore instead of a hyphen.

The UI is refreshed two times: when the message is first inserted in the database, before the request to the server, and then when the server response is received and the sequential ID is obtained. At this moment, the message is updated. The UI reacts to this callings show first a message with the status "pending" and then a message with the status "sent". It is almost impossible to see the change of status as the server is local and takes almost no time to respond a request.

Testing the App

In order to test the app, it is necessary to first run the server first. The command used to run the server during the tests was: **java -jar ChatServerSimple.jar localhost 81**

The above command sets the server name and port. It is important to keep in mind that the port used in the command must be the same configured in the app.With the app opened, it is

necessary to go to the settings first (the cog icon in the menu) and set the server address and username. The address must be set according with the settings used to run the server. The address used during the tests was: **http://10.0.2.2:81**

After the address and username are set, returning the the main screen will make a request to the server, registering the new user. This action will executed only once, as the app registration key and user id will be stored in the shared preferences of the app. After a successful registration, the "SEND" button becomes enabled and it is possible to send messages to the server.

Conclusion

The project APK can be found at the directory ***ChatServerWebServer\app\build\outputs\apk***

The video of the ChatApp with tasks running in the same process is available at the root folder with the name of ***simple_cloud_chatapp.swf***.